

Examples for commonly used methods

Table of contents:

- [Book a class](#)
- [Buy a membership](#)
- [Order and confirm a membership \(customer pays later\)](#)
- [Extend a membership](#)
- [Create a service booking](#)
- [Create a service booking and then cancel it](#)
- [Order a value card](#)
- [List service bookings where I am one of the resources](#)
- [Create an order with one or more products](#)
- [Create an order and pay with a discount card](#)
- [Create an order and add order items without specifying a customer](#)
- [Create and add items to a package booking](#)
- [Create an order with an item and a service booking](#)
- [Show the length of a waiting list and a specific position](#)

Requests with level 3 API key

- [Extend a membership with a request](#)
- [Use level 3 API for persons](#)
- [Create an order and add order items without specifying a customer](#)

Book a class

1. List all class/group activity products
[Products](#)

```
GET:APIURL/products.xml?apikey=338934897438&producttype=groupActivity
```

This will return a list of {productid}.

2. List classes/group activities
[Group activities/classes](#)

```
GET:APIURL/activities.xml?apikey=338934897438&businessunitids=4,334&product={productid} (Can be several id's)
```

This will return {activityid}

3. Create a class/group activity booking

[Class/group activity bookings](#)

```
POST:APIURL/activitybookings.xml
apikey=338934897438
type=ordinary
activityid={activityid}
```

Buy a subscription

1. List subscription products
[Products](#)

```
GET:APIURL/products.xml?apikey=338934897438&businessunitid=1&producttype=subscription
```

This will return {productid}

2. Create an order
[Orders](#)

```
POST:APIURL/orders.xml
apikey=338934897438
```

This will return {orderid}

3. Create a subscription booking and place it in the order
[order rows](#)

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
productid={productid}
```

This will return {lefttopay}

4. Pay for an order, create a receipt
[Receipts](#)

```
POST:APIURL/receipts.xml
apikey=338934897438
orderid={orderid}
payment.totalamount={lefttopay}
payment.paymentmethodid=1
```

If the payment is completed with a 3rd level API key the persons ID must be sent as "customerid".

Order and confirm a membership (customer pays later)

1. List membership products
[Products](#)

```
GET:APIURL/products.xml?apikey=338934897438&businessunitd=1&producttype=subscription
```

This will return {productid}

2. Create an order and confirm it, i.e. set preliminary to false. If this is not set the order will be considered abandoned and deleted after a preset time limit.
[Orders](#)

```
POST:APIURL/orders.xml
apikey=338934897438
preliminary=false
```

This will return {orderid}

3. Create a membership booking and place it in the order
[Order rows](#)

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
productid={productid}
```

Extend a membership

Extend the membership by one month. If the specified membership has a different debit interval an error message will be returned.

1. Create order
[Orders](#)

```
POST:APIURL/orders.xml
apikey=338934897438
```

This will return {orderid}

2. Retrieve membership ID for the logged on user

```
GET:APIURL/subscriptions.xml
apikey=338934897438
```

This will return {subscriptionid}

3. Create a membership booking and place it in the order

[Order rows](#)

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
subscriptionid={subscriptionid}
```

This will return {lefttopay}

4. Pay for an order, create receipt

[Receipt](#)

```
POST:APIURL/receipts.xml
apikey=338934897438
orderid={orderid}
payment.totalamount={lefttopay}
payment.paymentmethodid=1
```

Create a service booking

1. List services

[Products](#)

```
GET:APIURL/products.xml?apikey=338934897438&businessunitd=1&producttype=service
```

This will return {productid}, {resourcerequirementid} och {resourcerequirementtype}

2. List resources

[Resurser](#)

```
GET:APIURL/resources.xml?apikey=338934897438&businessunitid=1&type={resourcerequirementtype}
```

This will return {resourceid}

3. List available times

[Available times](#)

This will return {startdate} och {starttime}

4. Create order

[Orders](#)

```
POST:APIURL/orders.xml
apikey=338934897438
```

This will return {orderid}

5. Create a service booking and place it in the order

[Order rows](#)

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
productid={productid}
resource0.requirementid={resourcerequirementid}
resource0.id={resourceid}
startdate={startdate}
starttime={starttime}
```

This will return {lefttopay}

6. Pay an order, create a receipt
[Receipt](#)

```
POST:APIURL/receipts.xml
apikey=338934897438
orderid={orderid}
payment.totalamount={lefttopay}
payment.paymentmethodid=1
```

Create a service booking and then cancel it

1 - 5 Same as [Create a service booking](#)

(6. Confirm order)

Set preliminary to false. If this is not set the order will be considered abandoned and deleted after a preset time limit.

[Orders](#)

```
PUT:APIURL/orders/{orderid}.xml?apikey=338934897438&preliminary=false
```

7. Cancel service booking
[Orders](#)

```
DELETE:APIURL/orders/{orderid}.xml?apikey=338934897438
```

Order a value card

The same procedure as [Buy a membership](#) but search for `producttype=valueCard` in step 1 and specify `amount` in step 3.

List service bookings where I am one of the resources

Alternative #1:

1. List resources
[Resources](#)

```
GET:APIURL/resources.xml?apikey=338934897438&businessunitid=1&type=Personal
```

This will return {resourceid} where [resource.employee.id](#) is the same for the logged on user

2. List service bookings
[Service bookings](#)

```
GET:APIURL/servicebookings.xml?apikey=338934897438&businessunitids=1&resourceid={resourceid}
```

Alternative #2:

1. List service bookings
[Service bookings](#)

```
GET:APIURL/servicebookings.xml?apikey=338934897438&businessunitids=1&showallocated=true
```

Create an order with one or more products

1. List business units/facilities

[Business units/Facilities](#)

2. List products

[Products](#)

3. Verify user

[Verify person](#)

4. Create order

[Orders](#)

5. Add product to order

[Order rows](#)

Use the Id from step 4 and one of the products obtained in step 2.

Example for an item:

```
POST:APIURL/orders/{orderid}/items.xml
apikey:338934897438
productid={productid}
quantity=1
```

To add more products to an order repeat the above procedure.

6. Create an external payment

For example, by using an integration with a payment service provider (PSP)

7. Create a receipt for a payment

[Receipts](#)

Example:

```
POST:APIURL/receipts.xml
apikey:338934897438
orderid={orderid}
payment.totalamount={värdet på betalningen i ören}
payment.paymentmethodid=1
```

8. Display receipt

[Receipts](#)

```
GET:APIURL/receipts/{receiptid}.xml?apikey:338934897438
```

Create an order and pay with a discount card

1. List products

[Products](#)

```
GET:APIURL/products.xml?apikey=338934897438&businessunitd=1
```

This returns {productid}

2. List discount cards valid for the specified product

<link>Discount cards</link>

```
GET:APIURL/valuecards.xml?apikey=338934897438&productid={productid}
```

This returns {valuecardnumber}

3. Create order

[Orders](#)

```
POST:APIURL/orders.xml
apikey=338934897438
```

This returns {orderid}

4. Create a booking and enter it into an order

[Order rows](#)

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
productid={productid}
quantity=1
```

This returns {lefttopay}

5. Pay an order, create a receipt

[Receipts](#)

```
POST:APIURL/receipts.xml
apikey=338934897438
orderid={orderid}
payment.valuecardnumber={valuecardnumber}
payment.totalamount={lefttopay}
payment.paymentmethodid=6
```

Show the length of a waiting list and a specific position

There are two alternatives as per below:

Alternative 1:

List group activities/classes and include the bookings

[Group activities/classes](#)

```
GET:APIURL/activities.xml?apikey=338934897438&businessunitids=4,332&includebooking=true
```

Alternative 2:

List group activity/class bookings by waitinglist type

[Group activity/class bookings](#)

```
GET:APIURL/activitybookings.xml?apikey=338934897438&type=waitinglist
```

Create an order and add order items without specifying a customer

1. [Create Anonymous Order](#)

2. [Add an order item without specifying a customer](#)

3. Create a customer for the order or select an existing customer.

4. Add the customer to the order via the [change order](#) method but send the orderedbyid=<customerID>. Please note that due to rebates and pricelists for a specific customer that the order value may change. The value returned in this step is to be used in step 4.

5. Pay for the order by accessing receipts. If no customer is specified the parameter "anonymous=true" must be sent.

Note: Step 3 and 4 listed above are required if any of the order rows are subscriptions/memberships.

Create and add items to a package booking

1. List packages

[Products](#)

Examples of packages

```
GET:APIURL/products.xml?apikey=338934897438&producttype=package
```

The response received for a package will also contain individual package items (see [products](#)).

2. Create an order

```
POST:APIURL/orders.xml
apikey=338934897438
```

3. Create a package booking

Create an orderrow with productid = package product ID and quantity. The price for this order row is 0. Please note the item.id for the specific order row

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
productid=<package product id>
quantity=1
```

4. Fill the package with individual package items (packageparts)

Create order rows with products that are included in the package part by product type. Bear in mind that each type of product (services, item etc.) requires different parameters (see [items](#)).

Ad the packageid parameter with the value that the package orderrow was given.

```
POST:APIURL/orders/{orderid}/items.xml
apikey=338934897438
productid=99
quantity=1
packageid = <item.id> from 3.
```

BRP performs a sanity check that the specific products are in fact valid for the package product. Other requirements are left up to the integrating developer.

5. Complete purchase as usual

Create an order with an item and a service booking

Follow step 1 through 7 located at [Create an order with one or more products](#)

1. List services

[Products](#)

Examples:

```
GET:APIURL/products.xml?apikey: 338934897438 &producttype=service
```

2. Show available times

Use one of the products obtained with step 6

[Available times](#)

3. Book service

[Order rows](#)

Use the ID obtained in step X and a starting time from step 7. **Använd det id som erhölls i steg och en starttid som erhölls i steg 7.**

4. Show order

[Orders](#)

5. Create external payment

For example using an integration to a payment service provider (PSP)

6. Create receipt for payment

[Receipts](#)

Example:

```
POST:APIURL/receipts.xml
apikey: 338934897438
orderid={orderid}
payment.totalamount={value of payment in cents/ören}
payment.paymentmethodid=1
```

7. Show receipt

[Receipt](#)

```
GET:APIURL/receipts/{receiptid}.xml?apikey: 338934897438
```